

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

College of Engineering and Technology

SRM Institute of Science and Technology

MINI PROJECT REPORT

ODD Semester, 2024-25



Lab code & Name : 18CSC303J – Database Management Systems

Year & Semester : IV and VII

Project Title : ONLINE FOOD MANAGEMENT SYSTEM

Lab Supervisor : **Sowmya N**

Team Members : Mahek Tiwari, Sakshi Singh, Deepshikha Singh

Reg. No.	RA21104301	RA21110430	RA211104301
Mark split-up	0085 Mahek Tiwari	10094 Sakshi Singh	0096 Deepshikha Singh
Novelty in the project work (2 marks)			
Level of understanding (4 marks)			
Contribution to the project (2 Marks)			
Report writing (2 Marks)			
Total (10 Marks)			

Date:

Signature of Lab

Supervisor

Index

- TITLE:
- OBJECTIVE:
- ABSTRACT:
- INTRODUCTION:
- BLOCK DIAGRAM/ INTERFACE DIAGRAM:
- ALGORITHM
- APPROACH/METHODOLOGY:
- REALISTIC CONSTRAINTS:
- RESULTS AND DISCUSSION:
- CONCLUSION:
- REFERENCES:
- APPENDIX:
- Code:

ONLINE FOOD MANAGEMENT SYSTEM

OBJECTIVE:

This project is a food shop that allows users to check for different food items available at the online shop and then purchase online.

ABSTRACT:

The project provides a list of food products displayed online in various categories. The user may browse through these items. If the user wants to purchase any product(s), he/she may add it to his shopping cart.

Keeping the features of an e-commerce site, an online bakery shop software project acts as a central database containing various bakery products. It provides customers online shopping facilities from their homes.

A customer can sign up for free, log in to his / her account can browse items of his / her own interest, and can view prices and other details of selected items, place items with preferred weights into the shopping cart and can select from payment options. After that, the user can check out. The user can select any payment option that he/she wish to, like through credit/debit card or cash on delivery.

INTRODUCTION:

Keeping the features of an e-commerce site, an online bakery shop software project acts as a central database containing various bakery products. It provides customers online shopping facilities from their homes.

A customer can sign up for free, log in to his / her account can browse items of his / her own interest, and can view prices and other details of selected items, place items with preferred weights into the shopping cart and can select from payment options. After that, the user can check out. The user can select any payment option that he/she wish to like through credit/debit card or cash on delivery.

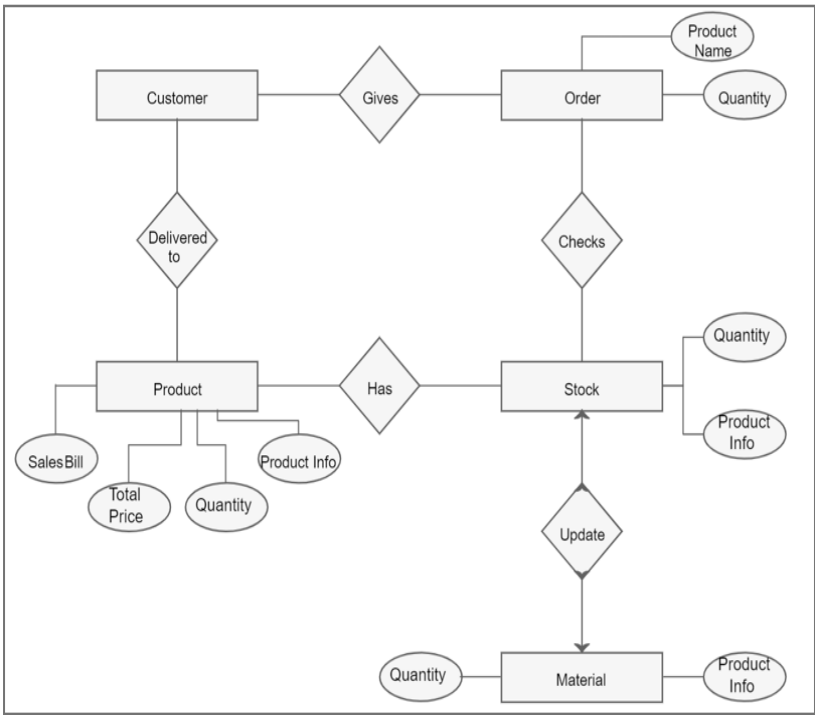
Keeping the features of an e-commerce site, an online shop software project acts as a central database containing various products. It provides customers online shopping facilities from their homes.

A customer can sign up for free, log in to his / her account can browse items of his / her own interest, and can view prices and other details of selected items, place items with preferred weights into

the shopping cart and can select from payment options. After that, the user can check out. The user can select any payment option that he/she wish to, like through credit/debit card or cash on delivery.

The **Cashiers** are only allowed to manage transactions and list the sales report of their transactions. This has a Point of Sale feature to manage the shop’s daily sales. It has also a feature that generates a printable receipt and date-wise sales report.

BLOCK DIAGRAM/ INTERFACE DIAGRAM:



Food Management System entities and their attributes:

Product Details: Used to store all main product items (Ex: Cake)

Stock Details: Used to store all items details which is needed to make main product items. (Ex: To make Cake we need Sugar, Milk etc...)

Transaction details

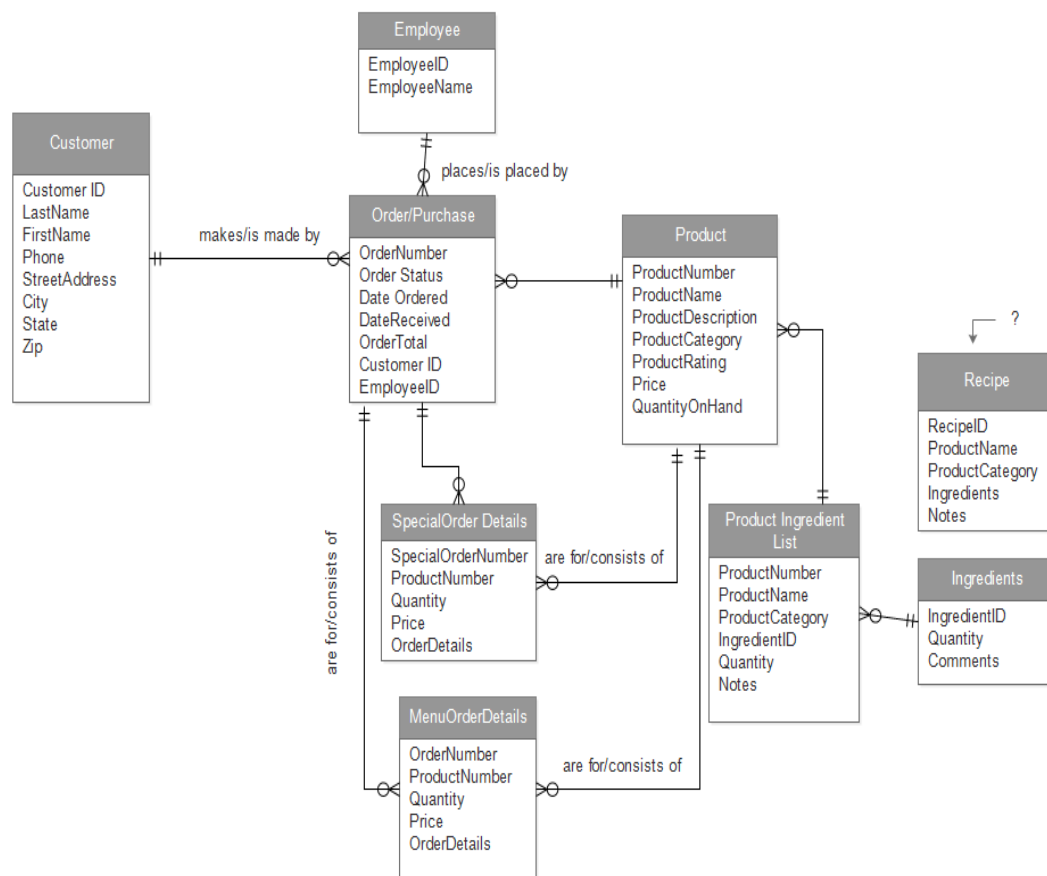
Transaction items

Billing Details

Admin details

Employee details

RELATIONAL TABLE DIAGRAM: -



ALGORITHM /SOFTWARE REQUIRED

Frontend: -

There are many frameworks and libraries available for frontend development.

We have used HTML, CSS, Ajax, Bootstrap for the frontend development.

HTML (Hyper Text Markup Language) is the most basic building block of the Web. It defines the meaning and structure of web content. It was used in the frontend part of our project.

Cascading Style Sheets is a style sheet language used for describing the presentation of a document written in a markup language such as HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

Ajax is a set of web development techniques that uses various web technologies on the client-side to create asynchronous web applications. With Ajax, web applications can send and retrieve data from a server asynchronously without interfering with the display and behavior of the existing page.

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains HTML, CSS and JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components

Backend: -

We have used PHP, MySQL, JavaScript framework for the backend

DESIGN ISSUES:

1. User Interface (UI) Design

Simplicity: Ensure the UI is simple and intuitive for all user types, including customers, restaurant staff, and administrators.

Responsiveness: The UI should be responsive to work seamlessly on different devices (desktops, tablets, and smartphones).

Accessibility: Follow accessibility standards to make the system usable for people with disabilities.

2. User Authentication and Authorization

Authentication: Implement a secure login system for different user roles (e.g., customers, staff, admins).

Authorization: Ensure users have appropriate permissions to access specific parts of the system based on their roles.

3. Database Design

Normalization: Use normalization techniques to avoid data redundancy and ensure data integrity.

Scalability: Design the database schema to accommodate future growth in data volume and user base.

Relationships: Define clear relationships between entities (e.g., users, orders, menu items, restaurants).

4. Order Management

Real-time Processing: Ensure orders are processed in real-time, updating the system promptly.

Order Tracking: Allow customers and staff to track the status of orders from placement to delivery.

Notifications: Implement notification mechanisms (e.g., email, SMS) for order confirmations, status updates, and delivery.

5. Menu Management

Dynamic Updates: Allow restaurant staff to update the menu, including prices, descriptions, and availability.

Categories: Organize menu items into categories for easy navigation.

Special Offers: Provide a way to manage special offers and discounts.

6. Payment Integration

Multiple Payment Methods: Support various payment methods, including credit/debit cards, digital wallets, and cash on delivery.

Security: Ensure secure payment processing using encryption and compliance with relevant standards (e.g., PCI DSS).

7. Data Security

Encryption: Encrypt sensitive data both in transit and at rest.

Backup: Implement regular data backup procedures to prevent data loss.

Access Control: Ensure only authorized users can access sensitive information.

8. Performance Optimization

Caching: Use caching mechanisms to reduce database load and improve response times.

Load Balancing: Implement load balancing to handle high traffic and ensure system reliability.

Database Indexing: Use indexing to speed up query performance.

9. Scalability and Future Growth

Modular Design: Design the system in a modular fashion to allow easy addition of new features.

Cloud Integration: Consider using cloud services for storage, computing power, and scalability.

10. Testing and Quality Assurance

Unit Testing: Write unit tests for individual components to ensure they work correctly.

Integration Testing: Test the integration of different modules to ensure they work together seamlessly.

User Acceptance Testing: Conduct user acceptance testing to ensure the system meets user requirements.

11. User Feedback and Support

Feedback Mechanism: Provide a way for users to submit feedback and report issues.

Support: Offer customer support channels (e.g., chat, email, phone) to assist users with any problems.

By addressing these design issues, you can create a robust and user-friendly online food management system that meets the needs of all stakeholders.

APPROACH/METHODOLOGY:

LIST OF MODULES:-

Admin-Side

- **Secure Login/Logout**
- **Home Page**
 - Summary
 - Inventory/Available Stocks List
 - Restock Products
- **Maintenance**
 - Add New Category
 - List All Categories
 - View Category Details
 - Update Category Details
 - Update Category Status
 - Delete Category
- **Manage Product**
 - Add New Product
 - List All Products
 - View Product Details
 - Update Product Details
 - Delete Product
- **Manage Stocks**
 - Add New Stock
 - List All Stocks
 - View Stock Details
 - Update Stock Details
 - Delete Stock
- **Point of Sale**
- **Generate Printable Receipt**
- **Generate Printable Date-wise Sales Report**
- **Manage User List (CRUD)**
- **Manage Account Credentials**

Admin-Side

- **Secure Login/Logout**
- **Home Page**
 - Summary
 - Inventory/Available Stocks List
- **Point of Sale**
- **Generate Printable Receipt**
- **Generate Printable Date-wise Sales Report of their Transactions**

Manage Account Credentials

The system requires user credentials in order for the management can access the data, features, and functionalities. It has 2 types of user roles which are the **Administrators** and **Cashiers**. The Administrator Users are the ones who can manage the products, stocks, and categories. They also have the privilege to access and manage all the features and functionalities of what the system does. The Cashiers are only allowed to manage transactions and list the sales report of their transactions. This has a Point of Sale feature to manage the shop's daily sales. It has also a feature that generates a printable receipt and date-wise sales report.

RELATIONAL SCHEMAS

1] Table 1: tblcustomer

Field Name	Description	Type	Length
Customer_ID	Customer ID number	int	10
Fname	Customer Fname	Varchar	10
Lname	Customer Lname	Varchar	10
Purchase Produ	Product Purchase of the Customer	Varchar	10

2] Table 2: tblproduct

Field Name	Description	Type	Length
------------	-------------	------	--------

code	Product ID number	int	10
Product Name	Name of the Product	Varchar	10
Alert	Stock needed	Int	10
Description	The Description of the Product	Text	30
Price	Price of the Product	int	10
Action	Delete, edit, add items	int	10

3: tblsales Table

Field Name	Description	Type	Length
Sales_ID	Sales ID number	int	10
Customer ame	Name of the Customer	Varchar	15
Product Name	Name of the Product	Varchar	15
Price	Price of the Product	int	10

Date Purchase	Date of the Transaction	int	10
Quantity	Number of item purchase	int	10

REALISTIC CONSTRAINTS:

The proposed system is designed to help the administrator to meet the demand of goods by sending and/or serving the request for food as and when required. The proposed system gives the procedural approach of how to bridge the gap between Recipient, sender, and shop. This Application will provide a common ground for all the three parties and will ensure the fulfillment of demand. The proposed system consists of the following goals and has the scope as follows:

a) Goals:

- To ease the process of delivery
- To improve the existing system.
- To develop a scalable system.
- To be highly available

b) Scope:

- Ensure that all the functionalities of a bakery are covered
- Make sure the program is simple and easy to use

RESULTS AND DISCUSSION:

With the food ordering system online, people can easily track their orders, and admin can maintain customer's database and advance the food delivery system. This food ordering system allows the user to select the desired food items from a list of available menu items provided by the local hotel or restaurant.

Advantages of the Online Food Ordering System

1. **Variety of Food**
2. **Convenience**
3. **Less Cost of Ad**
4. **Low Maintenance**
5. **Time-Savvy**
6. **Increase Loyalty**
7. **More Customers**
8. **Increase Visibility**

PRODUCTS

Product List

Add New

Show 10 entries

Search:

#	Code	Category	Product	Price	Alert	Status	Action
1	987545	Pastry	Black forest Love chocolate? Then this tender, luxurious layer pastry is for you. With a ganache and fabulous bittersweet filling, this indulgence is worth to try!	55	30	Active	Action
2	23141506	Bread	Bread Brown, multigrain	25	15	Active	Action
3	609809	Patties	Chocolate Coconut Cream Patties These soft, chewy chocolate-covered patties are bursting with sweet coconut flavor	35	40	Active	Action
4		Pastry	Chocolate Truffle a chocolate ganache centre coated in chocolate, cocoa powder, coconut, or chopped and toasted nuts.	60	15	Active	Action
5	568901	Pastry	Pineapple Pastry Sponge cake layered with cream and chopped pineapple, topped with whipped cream makes for this delicious dessert.	50	10	Active	Action

STOCKS

Stock List

Add New

Show 10 entries

Search:

#	Date Added	Product	Quantity	Expiry	Action
1	2022-04-21	23141506 Bread	3	-0001-11-30	Action
2	2022-03-17	23141506 Bread	2	-0001-11-30	Action
3	2022-02-14	987545 Black forest	100	2022-02-18	Action
4	2022-02-14	609809 Chocolate Coconut Cream Patties	350	2022-02-18	Action
5	2022-02-14	568901 Pineapple Pastry	200	2022-02-18	Action
6	2022-02-14	23141506 Bread	20	2022-02-17	Action

Search

Category	Product Code	Product Name	Price	Available Quantity
Pastry	987545	Black forest	55	0
Bread	23141506	Bread	25	0
Patties	609809	Chocolate Coconut Cream Patties	35	0
Pastry		Chocolate Truffle	60	0
Pastry	568901	Pineapple Pastry	50	0

Qty	Product	Total
1	987545 Black forest (55)	55
10	23141506 Bread (25)	250
5	609809 Chocolate Coconut Cream Patties (35)	175
Sub-Total		480
Tax (12%)Inclusive		57.6
Total		480

USERS

Users List

[Add New](#)

Show entries

Search:

#	Name	Username	Type	Action
1	Dhanush	dh8908	Administrator	Action
2	Harish	cblake	Cashier	Action
3	Ritika Agarwal	ra3909	Cashier	Action

Showing 1 to 3 of 3 entries

Previous Next

MANAGE ACCOUNTS

Manage Account

Full Name

Administrator

Username

admin

New Password

Leave the New Password field blank if you don't want update your password.

Old Password

[Update](#)

CODE:

```
SELECT
ishname AS
ISHES_IN_R1

FROM MENU, RESTAURANT
WHERE Mrid = Rid AND Rid = 1;
--2
SELECT Ocid AS CUSTOMER_ID, COUNT(*) AS NUMBER_OF_ORDERS
FROM ORDERS
GROUP BY Ocid;
--3
/*ALTER TABLE APPLIES_ON DROP CONSTRAINT
pplies_on_acode_fkey;
ALTER TABLE APPLIES_ON ADD CONSTRAINT
pplies_on_acode_fkey FOREIGN key (Acode) REFERENCES
COUPON(Code) ON DELETE SET NULL;*/
SELECT *
FROM COUPON
WHERE NOW() > Exp_date;
--4
SELECT Rid, Rname
FROM RESTAURANT GROUP BY Rating;
--5
SELECT Oid_, Did, Dfname, Dlname
FROM DELIVERY_PERSON, ORDER_
WHERE Did = Odid;

--c1
ALTER TABLE DELIVERY_PERSON ADD COLUMN REMARKS
VARCHAR(20);
UPDATE DELIVERY_PERSON SET REMARKS =
CASE
WHEN Drating > 4 AND Drating <= 5 THEN 'EXPERT'
WHEN Drating > 3 AND Drating <= 4 THEN 'PROFICIENT'
WHEN Drating > 2 AND Drating <= 3 THEN 'COMPETENT'
WHEN Drating > 1 AND Drating <= 2 THEN 'SATISFACTORY'
WHEN Drating >= 0 AND Drating <= 1 THEN 'NOVICE'
END;

--C2
```

```

SELECT fname AS customer_name, Cphone as customer_phone,
Address as customer_address, Rname as restaurant_name,
phone as restaurant_phone, Raddress as restaurant_address
FROM (RESTAURANT JOIN (ORDER_ JOIN (CUSTOMER JOIN MAKES ON
id = Mcid) ON Oid_ = Moid) ON Rid = Orid)
WHERE Odid = 1;
--c3
SELECT Cid, Fname
FROM CUSTOMER
EXCEPT (SELECT Cid, FNAME FROM CUSTOMER, ORDERS WHERE Cid
Ocid);
--C4
SELECT Oid_, SUM(Iprice) as TOTAL_PRICE
FROM ORDER_ JOIN ITEMS ON Oid_ = Ioid
GROUP BY Oid_;
--C5

```

```

/*INSERT INTO CITEMS(Ciname, Ciprice)
SELECT Dishname, Mprice
FROM RESTAURANT JOIN MENU ON RID = MRID
WHERE Rid = 1 AND Dishname = 'Veg Extravaganza', ; */
CREATE OR REPLACE VIEW ORDER_TOTAL
AS SELECT Oid_, SUM(Iprice) as TOTAL_PRICE
FROM ORDER_ JOIN ITEMS ON Oid_ = Ioid
GROUP BY Oid_;
WITH TEMP(AVERAGE) AS
(SELECT AVG(TOTAL_PRICE) FROM ORDER_TOTAL)
SELECT Oid_, TOTAL_PRICE
FROM ORDER_TOTAL, TEMP
WHERE ORDER_TOTAL.TOTAL_PRICE > TEMP.AVERAGE;

```

```

CREATE USER ADMIN WITH PASSWORD 'ADMIN' CREATEDB;
--delivery person
CREATE OR REPLACE VIEW DELIVERY1 AS
SELECT fname AS customer_name, Cphone as customer_phone,
Address as customer_address, Rname as restaurant_name,
phone as restaurant_phone, Raddress as restaurant_address

```



```

FROM (RESTAURANT JOIN (ORDER_ JOIN (CUSTOMER JOIN MAKES ON
id = Mcid) ON Oid_ = Moid) ON Rid = Orid)
WHERE Odid = 1;
CREATE OR REPLACE VIEW DELIVERY2 AS
SELECT fname AS customer_name, Cphone as customer_phone,
Address as customer_address, Rname as restaurant_name,
phone as restaurant_phone, Raddress as restaurant_address
FROM (RESTAURANT JOIN (ORDER_ JOIN (CUSTOMER JOIN MAKES ON
id = Mcid) ON Oid_ = Moid) ON Rid = Orid)
WHERE Odid = 2;
--customer
CREATE or REPLACE VIEW customer1 as
select fname, lname, cphone, CAddress, Email
from customer
where cid = 1;
--RESTAURANT
CREATE or REPLACE VIEW RESTAURANT1 as
select Rname, Rphone, Cuisine, Raddress
from RESTAURANT
where Rid = 1;

```

```

CREATE TABLE food(
email varchar(50) PRIMARY KEY,
password varchar(30) NOT NULL,
customername varchar(50) NOT NULL,
address varchar(60) NOT NULL,
item VARCHAR(20) NOT NULL
);

```

```

CREATE TABLE food(
email varchar(50) PRIMARY KEY,
password varchar(30) NOT NULL,
customername varchar(50) NOT NULL,
address varchar(60) NOT NULL,
item VARCHAR(20) NOT NULL
)

```

Table created.

```
INSERT INTO food(email,password,customername,address,item)
VALUES
```

```
('ramesh@gmail.com','hrishikesh13','Ramesh','Tamil Nadu', 'pastry'),
('abner@gmail.com','hrishikesh13','Ramesh','Tamil Nadu', 'pastry'),
('ashu@gmail.com','12345','sheetal','hyderabad', 'choco chips'),
('jatin@gmail.com','13','abhi','odisha', 'burger');
Select * from food;
```

1 row(s) inserted.

EMAIL	PASSWORD	CUSTOMERNAME	ADDRESS	ITEM
ramesh@gmail.com	hrishikesh13	Ramesh	Tamil Nadu	pastry
abner@gmail.com	hrishikesh13	Ramesh	Tamil Nadu	pastry
ashu@gmail.com	12345	sheetal	hyderabad	choco chips
jatin@gmail.com	13	abhi	odisha	burger

[Download CSV](#)

4 rows selected.

```
CREATE TABLE COUPON (
    Code VARCHAR(12),
    Maxdisc INT NOT NULL,
    Exp_date TIMESTAMP NOT NULL,
    Minord INT NOT NULL,
    PRIMARY KEY (Code)
);
```

```
CREATE TABLE food(
email varchar(50) PRIMARY KEY,
password varchar(30) NOT NULL,
customername varchar(50) NOT NULL,
address varchar(60) NOT NULL,
item VARCHAR(20) NOT NULL
)
```

Table created.

```
CREATE TABLE RESTAURANT (
```

```

Rid INT,
Rname VARCHAR(30) NOT NULL ,
Rphone VARCHAR(12) NOT NULL UNIQUE,
Raddress VARCHAR(200) NOT NULL UNIQUE,
Cuisine VARCHAR(20) NOT NULL,
Rating NUMERIC(2, 1),
Main_rid INT,
PRIMARY KEY (Rid),
FOREIGN KEY (Main_rid) REFERENCES RESTAURANT(Rid)

```

```

CREATE TABLE RESTAURANT (
  Rid INT,
  Rname VARCHAR(30) NOT NULL ,
  Rphone VARCHAR(12) NOT NULL UNIQUE,
  Raddress VARCHAR(200) NOT NULL UNIQUE,
  Cuisine VARCHAR(20) NOT NULL,
  Rating NUMERIC(2, 1),
  Main_rid INT,
  PRIMARY KEY (Rid),
  FOREIGN KEY (Main_rid) REFERENCES RESTAURANT(Rid)
)

```

Table created.

```

CREATE TABLE COUPONname (
  Code VARCHAR(12),
  Maxdisc INT NOT NULL,
  Exp_date TIMESTAMP NOT NULL,
  Minord INT NOT NULL,
  PRIMARY KEY (Code)
)

```

```

CREATE TABLE COUPONname (
  Code VARCHAR(12),
  Maxdisc INT NOT NULL,
  Exp_date TIMESTAMP NOT NULL,
  Minord INT NOT NULL,
  PRIMARY KEY (Code)
)

```

Table created.

```
CREATE or REPLACE VIEW RESTAURANT1 as
select Rname, Rphone, Cuisine, Raddress
from RESTAURANT
where Rid = 1
```

View created.

DML COMMANDS –

Alter Table RESTAURANT drop constraint restaurant_main_rid_fkey;

--add col in citems

ALTER TABLE CITEMS ADD COLUMN ciqty int default 1;

ALTER TABLE CART ADD COLUMN total_price float default 0;

--triggers

create function update_price() RETURNS trigger as \$\$ BEGIN

UPDATE CART SET total_price = total_price + NEW.Ciprice * NEW.ciqty WHERE Cart_id =
NEW.Ccart_id;

RETURN NEW;

END;

\$\$

LANGUAGE 'plpgsql';

CREATE FUNCTION reduce_price() RETURNS trigger as \$\$

BEGIN

UPDATE CART SET total_price = total_price - OLD.Ciprice*OLD.ciqty WHERE Cart_id =
OLD.Ccart_id;

RETURN OLD;

END;

\$\$

```
LANGUAGE 'plpgsql';

-- Creating triggers

CREATE TRIGGER insert_summary AFTER INSERT ON citems FOR EACH
ROW
EXECUTE PROCEDURE update_price();

CREATE TRIGGER delete_summary BEFORE DELETE ON citems FOR
EACH ROW
EXECUTE PROCEDURE reduce_price();

---MAIN TABLE INSERTIONS

--coupon

INSERT INTO COUPON VALUES('ugsdkabue',150,'2021-12-01 23:10:00',700);
INSERT INTO COUPON VALUES('uhdfiugfa',100,'2021-12-01 23:10:00',500);
INSERT INTO COUPON VALUES('ugsdkfhu9',50,'2021-12-01 23:10:00',300);
INSERT INTO COUPON VALUES('ugsdrhyiq',120,'2021-12-01 23:10:00',700);


--transaction

INSERT INTO TRANSACTION_ VALUES(1,2000,'COD');
INSERT INTO TRANSACTION_ VALUES(2,2500,'UPI');
INSERT INTO TRANSACTION_ VALUES(3,3000,'UPI');
INSERT INTO TRANSACTION_ VALUES(4,4000,'CARD');
INSERT INTO TRANSACTION_ VALUES(5,1000,'UPI');


--delivery person

INSERT INTO DELIVERY_PERSON
VALUES(0,'Dummy','Dummy','5','true',100.20,4.2,'Dummy');
```

```
INSERT INTO DELIVERY_PERSON  
VALUES(6,'Geetha','Anand','9087965745','true',097.12,4.8,'J P Nagar,Bangalore');
```

```
INSERT INTO DELIVERY_PERSON  
VALUES(7,'Adhitya','Suresh','9000965745','true',098.10,4.0,'Chruch Street,Bangalore');
```

```
INSERT INTO DELIVERY_PERSON VALUES(8,'Rohan','K','9087992345','true',096.11,3.9,'J P  
Nagar,Bangalore');
```

```
INSERT INTO DELIVERY_PERSON  
VALUES(9,'Keshav','Kumar','9087235745','true',097.12,4.5,'Netkallappa Circle,Bangalore');
```

```
INSERT INTO DELIVERY_PERSON VALUES(10,'Ankush','S  
J','9087000745','true',099.04,4.6,'Whitefield,Bangalore');
```

--customer

```
INSERT INTO CUSTOMER VALUES(1,'Manognya','Singuru','1-467/56(4),2nd street,M G  
Road,Bangalore','9835678234','manogyna@gmail.com');
```

```
INSERT INTO CUSTOMER VALUES(2,'Smriti','Tilak','123-38/45(13),5th  
cross,Basavangudi,Bangalore','9871237456','smruthi@gmail.com');
```

```
INSERT INTO CUSTOMER VALUES(3,'Shruvi','D','1-105/18(15),1st main ,2nd  
cross,Maroli,Bangalore','6366298767','shruvi@gmail.com');
```

```
INSERT INTO CUSTOMER VALUES(4,'Ross','Geller','2-908/23(45),4th  
cross,Malleshwaram,Bangalore','9807654321','ross@gmail.com');
```

```
INSERT INTO CUSTOMER VALUES(5,'Monica','G','8-23/12(7),9th cross,H S R  
Layput,Bangalore','9807659321','monica@gmail.com');
```

```
INSERT INTO CUSTOMER VALUES(6,'Rachel','Green','21-34/3(5),7th  
cross,Jayanagar,Bangalore','9437654321','rachel@gmail.com');
```

```
INSERT INTO CUSTOMER VALUES(7,'Joey','Tribiyani','16-8/19(4),5th cross,J P  
Nagar,Bangalore','9807654987','joey@gmail.com');
```

```
INSERT INTO CUSTOMER VALUES(8,'Chandler','Bing','9-8/3(16),8th  
cross,Whitefield,Bangalore','9065654321','chandler@gmail.com');
```

```
INSERT INTO CUSTOMER VALUES(9,'Ramesh','shetty','1-906/25(15),1st  
cross,Nelamangala,Bangalore','9873456723','ramesh@gmail.com');
```

INSERT INTO CUSTOMER VALUES(10,'arjun','Singh','12-89/5(1),5th cross,R R Nagar,Bangalore','9899956723','arjun@gmail.com');

--RESTAURANT

INSERT INTO RESTAURANT VALUES(1, 'Dominos', '9393939393',

'Ground Floor, 183, Old Outer Ring Rd, opp. Bangalore Development Authority Complex, 3rd Block, BDA Layout, 2nd Block, Naagarabhaavi, Bengaluru, Karnataka 560072',

'Italian', 4.0, 1);

INSERT INTO RESTAURANT VALUES(2, 'Dominos', '9393939392',

'Kemapura Dakale (Agrahara), XXI division, Bangalore city corporation, Nagarabhavi Main Rd, Govindaraja Nagar Ward, Prashant Nagar, Vijayanagar, Bengaluru, Karnataka 560079',

'Italian', 3.9, 1);

INSERT INTO RESTAURANT VALUES(3, 'Meghana Foods', '9393456921',

'52, 1st Floor, 33rd Cross Rd, near Cafe Coffee Day, 4th Block, Jayanagar, Bengaluru, Karnataka 560011',

'Andhra style', 4.3, 3);

INSERT INTO RESTAURANT VALUES(4, 'Meghana Foods', '9393456969',

'1st Cross Road, 124, 1st A Cross Rd, near Jyoti Nivas College, KHB Colony, 5th Block, Koramangala, Bengaluru, Karnataka 560095',

'Andhra style', 4.1, 3);

INSERT INTO MENU VALUES('Brownie Fantasy', 1, 59, 'Dessert', 'Sweet Temptation! Hot Chocolate Brownie drizzled with chocolate fudge sauce');

INSERT INTO MENU VALUES('Pepsi(500 mL)', 1, 60, 'Beverages', 'Sparkling and Refreshing Beverage');

INSERT INTO MENU VALUES('Margherita Pizza', 2, 99, 'Main Course', 'Classic delight with 100% real mozzarella cheese');

```
INSERT INTO MENU VALUES('Veg Extravaganza', 2, 249, 'Main Course', 'Black olives, capsicum, onion, grilled mushroom, corn, tomato, jalapeno & extra cheese');
```

```
INSERT INTO MENU VALUES('Cheese n Corn', 2, 169, 'Main Course', 'A delectable combination of sweet & juicy golden corn');
```

```
INSERT INTO MENU VALUES('Potato Cheese Shots', 2, 70, 'Appetizer', 'Crisp and golden
```

```
INSERT INTO MENU VALUES('Sweet Mango Lassi', 9, 89, 'Beverages', 'Mango lassi made with 100% natural mango pulp');
```

```
INSERT INTO MENU VALUES('Masala Butter Milk', 9, 79, 'Beverages', 'Spicy chilled buttermilk');
```

```
INSERT INTO ORDER_ VALUES(1, 2, 3);
```

```
INSERT INTO ORDER_ VALUES(2, 1, 3);
```

```
INSERT INTO ORDER_ VALUES(3, 7, 1);
```

```
INSERT INTO ORDER_ VALUES(4, 5, 2);
```

```
INSERT INTO ORDER_ VALUES(5, 4, 5);
```

```
INSERT INTO ITEMS VALUES('Chilly Gobi', 245, 1);
```

```
INSERT INTO ITEMS VALUES('Jeera Rice', 190, 1);
```

```
INSERT INTO ITEMS VALUES('Vanilla Icecream', 40, 2);
```

```
INSERT INTO ITEMS VALUES('Margherita Pizza', 99, 3);
```

```
--ORDERS
```

```
INSERT INTO ORDERS VALUES(1, 5);
```

```
INSERT INTO ORDERS VALUES(2, 6);
```

```
INSERT INTO ORDERS VALUES(2, 1);
```

```
UPDATE RESTAURANT SET rpassword = 'r1' WHERE rid = 1;
```



```
UPDATE RESTAURANT SET rpassword = 'r2' WHERE rid = 2;
```

```
UPDATE RESTAURANT SET rpassword = 'r3' WHERE rid = 3;
```

```
UPDATE RESTAURANT SET rpassword = 'r9' WHERE rid = 9;
```

```
ALTER TABLE ITEMS ADD COLUMN iqty int default 1;
```

```
ALTER TABLE RESTAURANT ADD COLUMN rreviews int default 1;
```

```
ALTER TABLE DELIVERY_PERSON ADD COLUMN dreviews int default 1;
```

```
--add review column
```

```
ALTER TABLE DELIVERY_PERSON ADD COLUMN REMARKS VARCHAR(20);
```

```
UPDATE DELIVERY_PERSON SET REMARKS =
```

```
CASE
```

```
WHEN Drating > 4 AND Drating <= 5 THEN 'EXPERT'
```

```
WHEN Drating > 3 AND Drating <= 4 THEN 'PROFICIENT'
```

```
WHEN Drating > 2 AND Drating <= 3 THEN 'COMPETENT'
```

```
WHEN Drating > 1 AND Drating <= 2 THEN 'SATISFACTORY'
```

```
WHEN Drating >= 0 AND Drating <= 1 THEN 'NOVICE'
```

```
END;
```

```
Alter Table RESTAURANT add constraint restaurant_main_rid_fkey FOREIGN KEY (Main_rid)  
REFERENCES RESTAURANT(Rid);
```

CONCLUSION:

To summarize, the world is rapidly evolving and heading toward technical expertise. Technology is not a static or stagnant field, but rather one that is constantly changing as new trends arise. As patterns change and improve, it's past time for us to change with them. The use of online food ordering management systems is important for getting accountability and making goods get delivered quickly and making the work easier. As a result, this system would make it easier for bakery companies be accountable for all their services

REFERENCES:

1. <http://www.mysqltutorial.org/>
2. <https://www.javatpoint.com/mysql-tutorial>

APPENDIX:

Database Schema and Design: The database schema of our Online Food Management System is designed to efficiently handle various operations, including user management, order processing, and menu management. The primary entities include Users, Orders, MenuItems, and Restaurants. Each entity is represented by a table in our MongoDB database, with appropriate fields and relationships. For instance, the Users table contains fields such as user_id, name, email, and address, while the Orders table links to both Users and MenuItems through foreign keys.

Data Flow and Functionality: The system allows users to browse the menu, place orders, and manage their accounts. The backend, built with Node.js, handles user requests and interactions with the MongoDB database. It ensures that orders are processed correctly and user data is securely managed. The frontend, developed using HTML and CSS, provides an intuitive interface for users to interact with the system.

